AF/ #

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

APPEAL BRIEF FOR THE APPELLANT

Ex parte Cheng-Liang (Andrew) HOU

SYSTEM AND METHOD FOR LINKING LIST TRANSMIT QUEUE MANAGEMENT

Serial No. 10/716,529
Appeal No.: TBD
Group Art Unit: 2619

Enclosed is a check in the amount of Five Hundred Ten Dollars ($510.00) to cover the official fee for this Appeal Brief. In the event that there may be any fees due with respect to the filing of this paper, please charge Deposit Account No. 50-2222.

Brad Y. Chin
Attorney for Appellant
Reg. No. 52,738

SQUIRE, SANDERS & DEMPSEY LLP
8000 Towers Crescent Drive, 14th Floor
Tysons Corner, VA 22182-6212

Atty. Docket: 058268.00325

BYC/dlh

Encls: Check No. 018934
Appeal Brief

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re the Appellant:

Cheng-Liang (Andrew) HOU	Appeal No.: TBD
Serial Number: 10/716,529	Group Art Unit: 2619
Filed: November 20, 2003	Examiner: Chu, Wutchung

For: SYSTEM AND METHOD FOR LINKING LIST TRANSMIT QUEUE MANAGEMENT

BRIEF ON APPEAL

June 4, 2008

I. INTRODUCTION

This is an appeal from the final rejection set forth in an Official Action dated December 11, 2007, finally rejecting claims 1-17, all of the claims pending in this application, as being unpatentable over Rusu, *et al.* (U.S. Patent No. 6,137,807). A Request for Reconsideration was timely filed on February 11, 2008. An Advisory Action was issued on March 6, 2008, indicating that the Request for Reconsideration had been considered, but did not place the application in condition for allowance. Therefore, claims 1-17 remain rejected. A Notice of Appeal and a Pre-Appeal Brief Request for Review were timely filed on March 11, 2008. A Notice of Panel Decision from Pre-Appeal Brief Review was issued on May 12, 2008, indicating that the rejections of claims 1-17 were maintained. Accordingly, this Appeal Brief is being timely filed.

II. REAL PARTY IN INTEREST

The real party in interest in this application is Broadcom Corporation of Irvine, California, by virtue of an Assignment by the inventors, which assignment was recorded at Reel 014730, Frame 0595, on November 20, 2003.

III. STATEMENT OF RELATED APPEALS AND INTERFERENCES

There are no known related appeals and/or interferences which will directly effect or be directly effected by or have a bearing on the Board's decision in this appeal.

IV. STATUS OF CLAIMS

Claims 1-17, all of the claims pending in the present application are the subject of this appeal. Claims 1-17 were rejected under 35 U.S.C. §102(b) as being allegedly anticipated by Rusu, *et al.* (U.S. Patent No. 6,137,807) ("Rusu"). A response was filed on February 11, 2007, and was entered, but the response did not include any amendments to the claims.

V. STATUS OF AMENDMENTS

All of claims 1-17 stand as they were previously presented prior to the Final Office Action. No amendments were made after the final rejection. Thus, claims 1-17 are pending and their respective rejections of claims 1-17 are appealed.

VI. SUMMARY OF CLAIMED SUBJECT MATTER

Claim 1, upon which claims 2-7 are dependent, recites a method (See Specification, at least at page 10, line 31, to page 12, line 12). The method includes receiving a packet (See Specification, at least at page 10, line 34, to page 11, line 1, step 510), determining an address of a free entry in a queue (See Specification, at least at page 11, lines 1-5, step 520), placing the determined address in an entry of a prior-determined address in the queue to form a linking list (See Specification, at least at page 11, lines 7-14, step 530), and placing packet data of the packet in a free entry of a first data structure (See Specification, at least at page 11, lines 7-14, steps 540 and 550). There is one-to-one mapping between the queue and the first data structure (See Specification, at least at page 11, line 31, to page 12, line 12).

Claim 8 recites a transmit queue system (Specification, at least on page 5, line 9, to page 6, line 23, linking list queue management system 170). The transmit queue system includes means for receiving a packet (Specification, at least on page 5, line 9, to page 6, line 23, packet receiving engine 205), means for determining an address of a free entry in a queue (Specification, at least on page 6, line 25, to page 7, line 2, and page 7, line 32, to page 8, line 12, free entry engine 310), means for placing the determined address in an entry of a prior-determined address in the queue to form a linking list (Specification, at least on page 8, line 32, to page 9, line 13, TXQ engine), and means for placing packet data of the packet in a free entry of a first data structure (Specification, at least on page 8, lines 24-30, BT engine 340 and BP engine 350). The transmit queue system also includes one-to-one mapping between the queue and the first data structure

(Specification, at least on page 7, lines 17-30).

Claim 9 recites a transmit queue system (Specification, at least on page 5, line 9, to page 6, line 23, linking list queue management system 170). The transmit queue system includes a first data structure capable of holding a plurality of packet data (Specification, at least on page 6, line 25, to page 7, line 2, memory manager 240), a queue capable of holding a linking list of addresses (Specification, at least on page 6, line 25, to page 7, line 2, transmit queue 215), a packet receiving engine capable of receiving a packet (Specification, at least on page 5, line 9, to page 6, line 23, packet receiving engine 205), a free entry engine coupled to the packet receiving engine and capable of determining an address of a free entry in the queue (Specification, at least on page 6, line 25, to page 7, line 2, and page 7, line 32, to page 8, line 12, free entry engine 310), a transmit queue engine (Specification, at least on page 8, line 32, to page 9, line 13, TXQ engine), and a packet buffer engine (Specification, at least on page 8, lines 24-30, BP engine 350). The addresses have a one-to-one mapping with addresses in the first data structure (Specification, at least on page 7, lines 17-30). The transmit queue engine is coupled to the queue, the packet receiving engine, and the free entry engine and is capable of placing the determined address in an entry of a prior-determined address in the queue to form a linking list (Specification, at least on page 8, line 32, to page 9, line 13). The packet buffer engine is coupled to the first data structure, the packet receiving engine, and the free entry engine and is capable of placing packet data of the packet in a free entry of the first data structure (Specification, at least on page 8, lines 24-30).

Claim 10, upon which claims 11-15 are dependent, recites a method (See Specification, at least at page 11, line 31, to page 12, line 12). The method includes receiving an address in queue, reading packets from an entry from a first data structure with the same address as the received address, the queue, and the first data structure having one-to-one mapping, transmitting the packet data to a network node associated with the queue, reading a next address in the queue from the received address in the queue, and using the next address to repeat the reading packet data and the transmitting (See Specification, at least at page 11, line 31, to page 12, line 12).

Claim 16 recites a transmit queue system (Specification, at least on page 5, line 9, to page 6, line 23, linking list queue management system 170). The transmit queue system includes means for receiving an address in a queue (Specification, at least on page 9, line 26, to page 10, line 2), means for reading packet data from an entry from a first data structure with the same address as the received address (Specification, at least on page 10, lines 4-15), means for transmitting the packet data to a network node associated with the queue (Specification, at least on page 6, line 25, to page 7, line 2, transmit queue 215), means for reading a next address in the queue from the received address in the queue (Specification, at least on page 10, lines 17-22), and means for using the next address to rerun the means for reading packet data and the means for transmitting (Specification, at least on page 10, lines 17-22). The queue and the first data structure have one-to-one mapping (Specification, at least on page 7, lines 17-30).

Claim 17 recites a transmit queue system (Specification, at least on page 5, line 9, to page 6, line 23, linking list queue management system 170). The transmit queue system includes a first data structure holding a plurality of packet data (Specification, at least on page 6, line 25, to page 7, line 2, memory manager 240), a queue holding a linking list of addresses (Specification, at least on page 6, line 25, to page 7, line 2, transmit queue 215), and a packet transmit engine (Specification, at least on page 8, line 32, to page 9, line 13, TXQ engine). The addresses have a one-to-one mapping with addresses in the first data structure (Specification, at least on page 7, lines 17-30). The packet transmit engine is coupled to the first data structure and the queue (Specification, at least on page 8, line 32, to page 9, line 13, TXQ engine). The packet transmit engine is capable of receiving an address in the queue, reading packet data from an entry from the first data structure with the same address as the received address, transmitting the packet data to a network node associated with the queue, reading a next address in the queue from the received address in the queue, and using the next address to repeat the reading packet data and the transmitting (Specification, at least on page 8, line 32, to page 9, line 13, TXQ engine).

VII. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

The grounds of rejection to be reviewed on appeal are the rejections of claims 1-17 under 35 U.S.C. §102(b) as being allegedly anticipated by Rusu.

VIII. APPELLANT'S ARGUMENTS

Appellant respectfully submits that each of pending claims 1-17 recites subject matter that is not taught, disclosed, or suggested by the cited art. Each of the claims is being argued separately under a separate sub-heading as suggested by 37 CFR 41.37(c)(1)(vii), and thus each of the claims stands or falls alone.

A. Claims 1-17 are novel in view of Rusu

In the Final Office Action of December 11, 2007, claims 1-17 were rejected under 35 U.S.C. §102(b) as being allegedly anticipated by Rusu ("Final Office Action"). Appellant respectfully submits that each of pending claims 1-17 recites subject matter that is not taught, disclosed, or suggested by Rusu, and as such, the Board's reversal or the rejections of claims 1-17 is respectfully requested.

1) Claim 1

Claim 1 recites a method. The method includes receiving a packet, determining an address of a free entry in a queue, placing the determined address in an entry of a prior-determined address in the queue to form a linking list, and placing packet data of the packet in a free entry of a first data structure. There is one-to-one mapping between the queue and the first data structure.

Appellant respectfully submits that claim 1 recites subject matter which is neither disclosed nor suggested by Rusu.

Rusu is directed to communication switches and data storage systems for use therewith. Rusu is also directed to a multiple queue bank balanced queue control system architecture. An input processor recognizes and accepts a wide variety of protocols and formats. Queue management stores uniform cells in a dual balanced bank memory system, which provides for utilizing an available bank of memory when the other bank of memory is in use, and otherwise balancing the use of the banks of memory, thereby maintaining equal free lists. Queue management apparatus and logic also ascertains and appends routing data to the stored data and transmits the data according to its priority (Rusu, Abstract; col. 1, lines 30-52).

Appellant notes that a "claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference" *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987). Additionally, the "identical invention must be shown in as complete detail as is contained in the...claim" *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989).

Appellant respectfully submits that the Final Office Action has failed to establish a *prima facie* case of anticipation as Rusu fails to disclose each and every element of claim 1. For example, Rusu fails to disclose or suggest, at least, "placing the determined address in an entry of a prior-determined address in the queue to form a linking list," as recited in claim 1.

Rusu, on the other hand, merely discloses a queue number, which is a 14 bit tag, indicating which queue memory bank (130, 131) a packet will be stored in (Rusu, col. 3, lines 35-37). Rusu further discloses that each internal cell as output by the input processors (101, 102) is assigned a queue number by the queue controller (140). The queue number is appended to the respective cells by the input processors (101, 102) and placed in a routing tag field within each cell (Rusu, col. 4, lines 11-16).

In Figure 5, Rusu further discloses that the queue controller (140) defines a memory location where the prepared cell will be sent based on the current queue memory bank (130, 131) availability, and if both banks (130, 131) are available, the bank chosen is that bank having the most available free memory space. If a memory bank (130 or 131) is currently in use, the queue controller (140) sends the prepared cell to the idle bank not in use. The dual bank queue memory system of Rusu allows for concurrent reading and writing to memory (Rusu, col. 4, lines 32-46).

The Final Office Action alleged that Rusu discloses "placing the determined address in an entry of a prior-determined address in the queue to form a linking list," citing Figure 5, box 365 – update link list for that queue, and column 4, lines 55-57 (See Final Office Action at page 3).

Rather, as previously noted, Rusu discloses appending the queue number in a routing tag field within each cell which is stored in an available queue memory bank (130, 131). Continuing with Figure 5, Rusu further discloses that data in the memory banks (130, 131) is maintained in the form of queues on a FIFO (first in, first out) basis, organized by a link list (102) maintained by the queue controller (140) with the control memory (145) (Rusu, col. 4, lines 47-57). Rusu fails to disclose or suggest that the queue number is placed in an entry of a prior-determined address in the queue to form a linking list. Rather, Rusu merely discloses that the link list (102) maintained by the queue controller (140) organizes data stored in the available queue memory banks (130, 131).

Hence, Rusu fails to disclose or suggest, at least, "placing <u>the determined address</u>" identified as the queue number in the Final Office Action, "in an <u>entry of a prior-determined address</u> in the queue to form a linking list," as recited in claim 1 (emphasis added).

Therefore, for at least the reasons discussed above, Appellant respectfully submits that Rusu fails to disclose or suggest, at least, "placing the determined address in an entry of a prior-determined address in the queue to form a linking list," as recited in claim 1.

Accordingly, Appellant respectfully asserts that the Final Office Action failed to establish a *prima facie* case of anticipation as Rusu fails to disclose each and every element of claim 1. As such, reconsideration and withdrawal of the rejection of claim 1 is respectfully requested.

2) Claim 2

Claim 2 is dependent upon claim 1, and recites further limitations. Accordingly, the arguments presented above for claim 1, apply with even greater force to claim 2.

Furthermore, claim 2 recites, "wherein the packet is unicast." Because Rusu fails to disclose the particular features recited in independent claim 1, Rusu also fails to disclose the features further defined in claim 2. Accordingly, it is respectfully requested that the rejection of claim 2 be reversed and claim 2 allowed.

3) Claim 3

Claim 3 is dependent upon claim 1, and recites further limitations. Accordingly, the arguments presented above for claim 1, apply with even greater force to claim 3.

Furthermore, claim 3 recites, "wherein the packet is multicast or broadcast, and further comprising: determining an address of a free entry in each queue associated with a destination in the packet; and for each queue associated with a destination in the packet, placing the respective determined address in a respective entry of a prior-determined address in each respective queue." Because Rusu fails to disclose the particular features recited in independent claim 1, Rusu also fails to disclose the features further defined in claim 3. Accordingly, it is respectfully requested that the rejection of claim 3 be reversed and claim 3 allowed.

4) Claim 4

Claim 4 is dependent upon claim 1, and recites further limitations. Accordingly, the arguments presented above for claim 1, apply with even greater force to claim 4.

Furthermore, claim 4 recites, "further comprising: determining a priority level for the received packet; and wherein the placing the determined address places the determined address in an entry of a prior-determined address in the queue having the same priority." Because Rusu fails to disclose the particular features for the prior-determined address in the queue recited in independent claim 1, Rusu also fails to disclose the features for the priority level further defined in claim 4. Accordingly, it is respectfully requested that the rejection of claim 4 be reversed and claim 4 allowed.

5) Claim 5

Claim 5 is dependent upon claims 1 and 4, and recites further limitations. Accordingly, the arguments presented above for claims 1 and 4, apply with even greater force to claim 5.

Furthermore, claim 5 recites, "wherein the determining a priority level includes examining a quality of service field within the received packet." Because Rusu fails to disclose the particular features for the prior-determined address in the queue recited in independent claim 1 and dependent claim 4, Rusu also fails to disclose the features for the priority level further defined in claim 5. Accordingly, it is respectfully requested that the rejection of claim 5 be reversed and claim 5 allowed.

6) Claim 6

Claim 6 is dependent upon claim 1, and recites further limitations. Accordingly, the arguments presented above for claim 1, apply with even greater force to claim 6.

Furthermore, claim 6 recites, "further comprising updating free entry data to indicate that the determined address is in use." Because Rusu fails to disclose the particular features recited in independent claim 1, Rusu also fails to disclose the features further defined in claim 6. Accordingly, it is respectfully requested that the rejection of claim 6 be reversed and claim 6 allowed.

7) Claim 7

Claim 7 is dependent upon claim 1, and recites further limitations. Accordingly, the arguments presented above for claim 1, apply with even greater force to claim 7.

Furthermore, claim 7 recites, "further comprising placing a packet length of the packet in a free entry of a second data structure; and wherein there is one-to-one mapping between the first data structure and the second data structure." Because Rusu fails to disclose the particular features recited in independent claim 1, Rusu also fails to disclose the features further defined in claim 7. Accordingly, it is respectfully requested that the rejection of claim 7 be reversed and claim 7 allowed.

8) Claim 8

Claim 8 recites a transmit queue system. The transmit queue system includes means for receiving a packet, means for determining an address of a free entry in a queue, means for placing the determined address in an entry of a prior-determined address in the queue to form a linking list, and means for placing packet data of the packet in a free entry of a first data structure. The transmit queue system also includes one-to-one mapping between the queue and the first data structure.

Appellant respectfully submits that claim 8 recites subject matter which is neither disclosed nor suggested by Rusu.

Rusu is directed to communication switches and data storage systems for use therewith. Rusu is also directed to a multiple queue bank balanced queue control system architecture. An input processor recognizes and accepts a wide variety of protocols and formats. Queue management stores uniform cells in a dual balanced bank memory system, which provides for utilizing an available bank of memory when the other bank of memory is in use, and otherwise balancing the use of the banks of memory, thereby maintaining equal free lists. Queue management apparatus and logic also ascertains and appends routing data to the stored data and transmits the data according to its priority (Rusu, Abstract; col. 1, lines 30-52).

Appellant respectfully submits that the Final Office Action has failed to establish a *prima facie* case of anticipation as Rusu fails to disclose each and every element of claim 8. For example, Rusu fails to disclose or suggest, at least, "means for placing the determined address in an entry of a prior-determined address in the queue to form a linking list," as recited in claim 8.

Rusu, on the other hand, merely discloses a queue number, which is a 14 bit tag, indicating which queue memory bank (130, 131) a packet will be stored in (Rusu, col. 3, lines 35-37). Rusu further discloses that each internal cell as output by the input processors (101, 102) is assigned a queue number by the queue controller (140). The queue number is appended to the respective cells by the input processors (101, 102) and placed in a routing tag field within each cell (Rusu, col. 4, lines 11-16).

In Figure 5, Rusu further discloses that the queue controller (140) defines a memory location where the prepared cell will be sent based on the current queue memory bank (130, 131) availability, and if both banks (130, 131) are available, the bank chosen is that bank having the most available free memory space. If a memory bank (130 or 131) is currently in use, the queue controller (140) sends the prepared cell to the idle bank not in use. The dual bank queue memory system of Rusu allows for concurrent reading and writing to memory (Rusu, col. 4, lines 32-46).

The Final Office Action alleged that Rusu discloses the "means for placing the determined address in an entry of a prior-determined address in the queue to form a linking list," citing column 3, lines 38-46, Figure 5, box 365 – update link list for that queue, and column 4, lines 55-57 (See Final Office Action at pages 5-6).

Rather, as previously noted, Rusu discloses appending the queue number in a routing tag field within each cell which is stored in an available queue memory bank (130, 131). Continuing with Figure 5, Rusu further discloses that data in the memory banks (130, 131) is maintained in the form of queues on a FIFO (first in, first out) basis, organized by a link list (102) maintained by the queue controller (140) with the control memory (145) (Rusu, col. 4, lines 47-57). Rusu fails to disclose or suggest that the queue number is placed in an entry of a prior-determined address in the queue to form a linking list. Rather, Rusu merely discloses that the link list (102) maintained by the queue controller (140) organizes data stored in the available queue memory banks (130, 131).

Hence, Rusu fails to disclose or suggest, at least, "means for placing <u>the determined address</u>" identified as the queue number in the Final Office Action, "in an <u>entry of a prior-determined address</u> in the queue to form a linking list," as recited in claim 8 (emphasis added).

Therefore, for at least the reasons discussed above, Appellant respectfully submits that Rusu fails to disclose or suggest, at least, "means for placing the determined address in an entry of a prior-determined address in the queue to form a linking list," as recited in claim 8.

Accordingly, Appellant respectfully asserts that the Final Office Action failed to establish a *prima facie* case of anticipation as Rusu fails to disclose each and every element of claim 8. As such, reconsideration and withdrawal of the rejection of claim 8 is respectfully requested.

9) <u>Claim 9</u>

Claim 9 recites a transmit queue system. The transmit queue system includes a first data structure capable of holding a plurality of packet data, a queue capable of holding a linking list of addresses, a packet receiving engine capable of receiving a packet, a free entry engine coupled to the packet receiving engine and capable of determining an address of a free entry in the queue, a transmit queue engine, and a packet buffer engine. The addresses have a one-to-one mapping with addresses in the first data structure. The transmit queue engine is coupled to the queue, the packet receiving engine, and the free entry engine and is capable of placing the determined address in an entry of a prior-determined address in the queue to form a linking list. The packet buffer engine is coupled to the first data structure, the packet receiving engine, and the free entry engine and is capable of placing packet data of the packet in a free entry of the first data structure.

Appellant respectfully submits that claim 9 recites subject matter which is neither disclosed nor suggested by Rusu.

Rusu is directed to communication switches and data storage systems for use therewith. Rusu is also directed to a multiple queue bank balanced queue control system architecture. An input processor recognizes and accepts a wide variety of protocols and formats. Queue management stores uniform cells in a dual balanced bank memory system, which provides for utilizing an available bank of memory when the other bank of memory is in use, and otherwise balancing the use of the banks of memory, thereby maintaining equal free lists. Queue management apparatus and logic also ascertains and appends routing data to the stored data and transmits the data according to its priority (Rusu, Abstract; col. 1, lines 30-52).

Appellant notes that a "claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference" *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987). Additionally, the "identical invention must be shown in as complete detail as is contained in the...claim" *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989).

Appellant respectfully submits that the Final Office Action has failed to establish a *prima facie* case of anticipation as Rusu fails to disclose each and every element of claim 9. For example, Rusu fails to disclose or suggest, at least, "a transmit queue engine, coupled to the queue, the packet receiving engine and the free entry engine, capable of placing the determined address in an entry of a prior-determined address in the queue to form a linking list," as recited in claim 9.

Rusu, on the other hand, merely discloses a queue number, which is a 14 bit tag, indicating which queue memory bank (130, 131) a packet will be stored in (Rusu, col. 3, lines 35-37). Rusu further discloses that each internal cell as output by the input processors (101, 102) is assigned a queue number by the queue controller (140). The queue number is appended to the respective cells by the input processors (101, 102) and placed in a routing tag field within each cell (Rusu, col. 4, lines 11-16).

In Figure 5, Rusu further discloses that the queue controller (140) defines a memory location where the prepared cell will be sent based on the current queue memory bank (130, 131) availability, and if both banks (130, 131) are available, the bank chosen is that bank having the most available free memory space. If a memory bank (130 or 131) is currently in use, the queue controller (140) sends the prepared cell to the idle bank not in use. The dual bank queue memory system of Rusu allows for concurrent reading and writing to memory (Rusu, col. 4, lines 32-46).

The Final Office Action alleged that Rusu discloses "a transmit queue engine ...capable of placing the determined address in an entry of a prior-determined address in the queue to form a linking list," citing Figure 1, box 160, output processor, Figure 4, box 147, output port link for q1 and q2, and Figure 7, link lists (See Final Office Action at page 7).

Rather, as previously noted, Rusu discloses appending the queue number in a routing tag field within each cell which is stored in an available queue memory bank (130, 131). Continuing with Figure 5, Rusu further discloses that data in the memory banks (130, 131) is maintained in the form of queues on a FIFO (first in, first out) basis, organized by a link list (102) maintained by the queue controller (140) with the control memory (145) (Rusu, col. 4, lines 47-57). Rusu fails to disclose or suggest that the queue number is placed in an entry of a prior-determined address in the queue to form a linking list. Rather, Rusu merely discloses that the link list (102) maintained by the queue controller (140) organizes data stored in the available queue memory banks (130, 131).

Hence, Rusu fails to disclose or suggest, at least, "a transmit queue engine... capable of placing the determined address" identified as the queue number in the Final Office Action, "in an entry of a prior-determined address in the queue to form a linking list," as recited in claim 9 (emphasis added).

Therefore, for at least the reasons discussed above, Appellant respectfully submits that Rusu fails to disclose or suggest, at least, "a transmit queue engine, coupled to the queue, the packet receiving engine and the free entry engine, capable of placing the determined address in an entry of a prior-determined address in the queue to form a linking list," as recited in claim 9.

Accordingly, Appellant respectfully asserts that the Final Office Action failed to establish a *prima facie* case of anticipation as Rusu fails to disclose each and every element of claim 9. As such, reconsideration and withdrawal of the rejection of claim 9 is respectfully requested.

10) Claim 10

Claim 10 recites a method of receiving an address in queue, reading packets from an entry from a first data structure with the same address as the received address, the queue, and the first data structure having one-to-one mapping, transmitting the packet data to a network node associated with the queue, reading a next address in the queue from the received address in the queue, and using the next address to repeat the reading packet data and the transmitting.

Appellant respectfully submits that claim 10 recites subject matter which is neither disclosed nor suggested by Rusu.

Rusu is directed to communication switches and data storage systems for use therewith. Rusu is also directed to a multiple queue bank balanced queue control system architecture. An input processor recognizes and accepts a wide variety of protocols and formats. Queue management stores uniform cells in a dual balanced bank memory system, which provides for utilizing an available bank of memory when the other bank of memory is in use, and otherwise balancing the use of the banks of memory, thereby maintaining equal free lists. Queue management apparatus and logic also ascertains and appends routing data to the stored data and transmits the data according to its priority (Rusu, Abstract; col. 1, lines 30-52).

Appellant notes that a "claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference" *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987). Additionally, the "identical invention must be shown in as complete detail as is contained in the...claim" *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989).

Appellant respectfully submits that the Final Office Action has failed to establish a *prima facie* case of anticipation as Rusu fails to disclose each and every element of claim 10. For example, Rusu fails to disclose or suggest, at least, "receiving an address in a queue; reading packet data from an entry from a first data structure with the same address as the received address...reading a next address in the queue from the received address in the queue; and using the next address to repeat the reading packet data and the transmitting," as recited in claim 10.

Rusu, on the other hand, merely discloses a queue controller which defines the memory location where a prepared cell will be sent based on a current queue memory bank (130, 131) availability. If both banks are available, the bank chosen is that having the most available free memory space. If a memory bank (130, 131) is currently in use, the queue controller (140) sends the prepared cell to the idle bank not in use. This dual bank queue memory system allows for concurrent reading and writing to memory, e.g. an individual or simultaneous reading and writing to and from the queue memory banks (130, 131) (Rusu, col. 4, lines 32-46).

Furthermore, Rusu discloses that queue controller (140) ensures load balancing between the two queue memory banks (130, 131) to balance the amount of available memory in each queue bank. Data in the memory banks (130, 131) is maintained in the form of queues on a FIFO basis, organized by a link list (102) maintained by the queue controller (140), as illustrated in Figure 2, with control memory (145) (Rusu, col. 4, lines 47-57).

Furthermore, Rusu discloses a cell in queue memory consisting of two parts, data and a control header. The control header is established by the queue controller and contains the link list pointers, or addresses, of the data, as illustrated in Figure 7. As shown in Figure 7, which illustrates a detailing control memory (145), the header portion (Q1, F1), may be stored in one queue memory location and the tail portion may be stored in another queue memory. The link list is used to enable this operation and keeps track of the mapping relationships (Rusu, col. 4, lines 58-67).

The Final Office Action alleged that Rusu discloses "receiving an address in a queue; reading packet data from an entry from a first data structure with the same address as the received address...reading a next address in the queue from the received address in the queue; and using the next address to repeat the reading packet data and the transmitting," citing column 4, lines 62-67, Figure 6, box 460, and column 5, lines 31-34 (See Final Office Action at pages 7-8).

However, a review of these passages and figures demonstrates that Rusu fails to disclose or suggest the aforementioned features recited in claim 10. For example, Rusu, at column 4, lines 62-67, fails to disclose or suggest, "receiving an address in a queue" (emphasis added). Rather, Rusu, as previously discussed above, merely discloses a cell in queue memory consisting of two parts, data and a control header. The control header is established by the queue controller and contains the link list pointers (or address) of the data, as illustrated in Figure 7. The header portion of a data cell may be stored in one queue memory location and the tail portion may be stored in another queue memory location. *The link list, or address, of the data enables this operation.* Hence, no *address* is received in the queue; rather, the link list, or the address, is used to keep track of and map relationships. Therefore, Rusu fails to disclose or suggest, at least, "receiving an address in a queue," as recited in claim 10 (emphasis added).

Accordingly, Rusu also fails to disclose or suggest, at least, "reading packet data from an entry from a first data structure with the same address as the received address...reading a next address in the queue from the received address in the queue; and using the next address to repeat the reading packet data and the transmitting," as recited in claim 10.

Therefore, for at least the reasons discussed above, Appellant respectfully submits that Rusu fails to disclose or suggest every feature recited in claim 10.

Accordingly, Appellant respectfully asserts that the Final Office Action failed to establish a *prima facie* case of anticipation as Rusu fails to disclose each and every element of claim 10. As such, reconsideration and withdrawal of the rejection of claim 10 is respectfully requested.

11) Claim 11

Claim 11 is dependent upon claim 10, and recites further limitations. Accordingly, the arguments presented above for claim 10, apply with even greater force to claim 11.

Furthermore, claim 11 recites, "further comprising reading packet length from a second data structure, the second data structure having one-to-one mapping with the first data structure." Because Rusu fails to disclose the particular features recited in independent claim 10, Rusu also fails to disclose the features further defined in claim 11.

Accordingly, it is respectfully requested that the rejection of claim 11 be reversed and claim 11 allowed.

12) Claim 12

Claim 12 is dependent upon claim 10, and recites further limitations. Accordingly, the arguments presented above for claim 10, apply with even greater force to claim 12.

Furthermore, claim 12 recites, "wherein the receiving receives an address for higher priority packet data before receiving an address for lower priority packet data." Because Rusu fails to disclose the particular features recited in independent claim 10, Rusu also fails to disclose the features further defined in claim 12. Accordingly, it is respectfully requested that the rejection of claim 12 be reversed and claim 12 allowed.

13) Claim 13

Claim 13 is dependent upon claim 10, and recites further limitations. Accordingly, the arguments presented above for claim 10, apply with even greater force to claim 13.

Furthermore, claim 13 recites, "wherein the packet data is multicast or broadcast and the transmitting transmits the packet data to a plurality of network nodes." Because Rusu fails to disclose the particular features recited in independent claim 10, Rusu also fails to disclose the features further defined in claim 13. Accordingly, it is respectfully requested that the rejection of claim 13 be reversed and claim 13 allowed.

14) Claim 14

Claim 14 is dependent upon claim 10, and recites further limitations. Accordingly, the arguments presented above for claim 10, apply with even greater force to claim 14.

Furthermore, claim 14 recites, "wherein the packet data is unicast." Because Rusu fails to disclose the particular features recited in independent claim 10, Rusu also fails to disclose the features further defined in claim 14. Accordingly, it is respectfully requested that the rejection of claim 14 be reversed and claim 14 allowed.

15) Claim 15

Claim 15 is dependent upon claim 10, and recites further limitations. Accordingly, the arguments presented above for claim 10, apply with even greater force to claim 15.

Furthermore, claim 15 recites, "further comprising updating free entry data to indicate an address is free after the transmitting." Because Rusu fails to disclose the particular features recited in independent claim 10, Rusu also fails to disclose the features further defined in claim 15. Accordingly, it is respectfully requested that the rejection of claim 5 be reversed and claim 15 allowed.

16) Claim 16

Claim 16 recites a transmit queue system. The transmit queue system includes means for receiving an address in a queue, means for reading packet data from an entry from a first data structure with the same address as the received address, means for transmitting the packet data to a network node associated with the queue, means for reading a next address in the queue from the received address in the queue, and means for using the next address to rerun the means for reading packet data and the means for transmitting. The queue and the first data structure have one-to-one mapping.

Appellant respectfully submits that claim 16 recites subject matter which is neither disclosed nor suggested by Rusu.

Rusu is directed to communication switches and data storage systems for use therewith. Rusu is also directed to a multiple queue bank balanced queue control system architecture. An input processor recognizes and accepts a wide variety of protocols and formats. Queue management stores uniform cells in a dual balanced bank memory system, which provides for utilizing an available bank of memory when the other bank of memory is in use, and otherwise balancing the use of the banks of memory, thereby maintaining equal free lists. Queue management apparatus and logic also ascertains and appends routing data to the stored data and transmits the data according to its priority (Rusu, Abstract; col. 1, lines 30-52).

Appellant notes that a "claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference" *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987). Additionally, the "identical invention must be shown in as complete detail as is contained in the...claim" *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989).

Appellant respectfully submits that the Final Office Action has failed to establish a *prima facie* case of anticipation as Rusu fails to disclose each and every element of claim 16. For example, Rusu fails to disclose or suggest, at least, "means for receiving an address in a queue; means for reading packet data from an entry from a first data structure with the same address as the received address... means for reading a next address in the queue from the received address in the queue; and means for using the next address to repeat the reading packet data and the transmitting," as recited in claim 16.

Rusu, on the other hand, merely discloses a queue controller which defines the memory location where a prepared cell will be sent based on a current queue memory bank (130, 131) availability. If both banks are available, the bank chosen is that having the most available free memory space. If a memory bank (130, 131) is currently in use, the queue controller (140) sends the prepared cell to the idle bank not in use. This dual bank queue memory system allows for concurrent reading and writing to memory, e.g. an individual or simultaneous reading and writing to and from the queue memory banks (130, 131) (Rusu, col. 4, lines 32-46).

Furthermore, Rusu discloses that queue controller (140) ensures load balancing between the two queue memory banks (130, 131) to balance the amount of available memory in each queue bank. Data in the memory banks (130, 131) is maintained in the form of queues on a FIFO basis, organized by a link list (102) maintained by the queue controller (140), as illustrated in Figure 2, with control memory (145) (Rusu, col. 4, lines 47-57).

Furthermore, Rusu discloses a cell in queue memory consisting of two parts, data and a control header. The control header is established by the queue controller and contains the link list pointers, <u>or addresses</u>, of the data, as illustrated in Figure 7. As shown in Figure 7, which illustrates a detailing control memory (145), the header portion (Q1, F1), may be stored in one queue memory location and the tail portion may be stored in another queue memory. The link list is used to enable this operation and keeps track of the mapping relationships (Rusu, col. 4, lines 58-67).

The Final Office Action alleged that Rusu discloses "means for receiving an address in a queue; means for reading packet data from an entry from a first data structure with the same address as the received address... means for reading a next address in the queue from the received address in the queue; and means for using the next address to repeat the reading packet data and the transmitting," citing column 4, lines 62-67, Figure 6, box 460, and column 5, lines 31-34 (See Final Office Action at pages 10-11).

However, a review of these passages and figures demonstrates that Rusu fails to disclose or suggest the aforementioned features recited in claim 16. For example, Rusu, at column 4, lines 62-67, fails to disclose or suggest, "means for receiving an address in a queue" (emphasis added). Rather, Rusu, as previously discussed above, merely discloses a cell in queue memory consisting of two parts, data and a control header. The control header is established by the queue controller and contains the link list pointers (or address) of the data, as illustrated in Figure 7. The header portion of a data cell may be stored in one queue memory location and the tail portion may be stored in another queue memory location. *The link list, or address, of the data enables this operation*. Hence, no *address* is received in the queue; rather, the link list, or the address, is used to keep track of and map relationships. Therefore, Rusu fails to disclose or suggest, at least, "means for receiving an address in a queue," as recited in claim 16 (emphasis added).

Accordingly, Rusu also fails to disclose or suggest, at least, "means for reading packet data from an entry from a first data structure with the same address as the received address... means for reading a next address in the queue from the received address in the queue; and means for using the next address to repeat the reading packet data and the transmitting," as recited in claim 16.

Therefore, for at least the reasons discussed above, Appellant respectfully submits that Rusu fails to disclose or suggest every feature recited in claim 16.

Accordingly, Appellant respectfully asserts that the Final Office Action failed to establish a *prima facie* case of anticipation as Rusu fails to disclose each and every element of claim 16. As such, reconsideration and withdrawal of the rejection of claim 16 is respectfully requested.

17) Claim 17

Claim 17 recites a transmit queue system. The transmit queue system includes a first data structure holding a plurality of packet data, a queue holding a linking list of addresses, and a packet transmit engine. The addresses have a one-to-one mapping with addresses in the first data structure. The packet transmit engine is coupled to the first data structure and the queue. The packet transmit engine is capable of receiving an address in the queue, reading packet data from an entry from the first data structure with the same address as the received address, transmitting the packet data to a network node associated with the queue, reading a next address in the queue from the received address in the queue, and using the next address to repeat the reading packet data and the transmitting.

Appellant respectfully submits that claim 17 recites subject matter which is neither disclosed nor suggested by Rusu.

Rusu is directed to communication switches and data storage systems for use therewith. Rusu is also directed to a multiple queue bank balanced queue control system architecture. An input processor recognizes and accepts a wide variety of protocols and formats. Queue management stores uniform cells in a dual balanced bank memory system, which provides for utilizing an available bank of memory when the other bank of memory is in use, and otherwise balancing the use of the banks of memory, thereby maintaining equal free lists. Queue management apparatus and logic also ascertains and appends routing data to the stored data and transmits the data according to its priority (Rusu, Abstract; col. 1, lines 30-52).

Appellant notes that a "claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference" *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987). Additionally, the "identical invention must be shown in as complete detail as is contained in the...claim" *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989).

Appellant respectfully submits that the Final Office Action has failed to establish a *prima facie* case of anticipation as Rusu fails to disclose each and every element of claim 16. For example, Rusu fails to disclose or suggest, at least, "a packet transmit engine...capable of receiving an address in a queue; reading packet data from an entry from a first data structure with the same address as the received address... reading a next address in the queue from the received address in the queue; and using the next address to repeat the reading packet data and the transmitting," as recited in claim 17.

Rusu, on the other hand, merely discloses a queue controller which defines the memory location where a prepared cell will be sent based on a current queue memory bank (130, 131) availability. If both banks are available, the bank chosen is that having the most available free memory space. If a memory bank (130, 131) is currently in use, the queue controller (140) sends the prepared cell to the idle bank not in use. This dual bank queue memory system allows for concurrent reading and writing to memory, e.g. an individual or simultaneous reading and writing to and from the queue memory banks (130, 131) (Rusu, col. 4, lines 32-46).

Furthermore, Rusu discloses that queue controller (140) ensures load balancing between the two queue memory banks (130, 131) to balance the amount of available memory in each queue bank. Data in the memory banks (130, 131) is maintained in the form of queues on a FIFO basis, organized by a link list (102) maintained by the queue controller (140), as illustrated in Figure 2, with control memory (145) (Rusu, col. 4, lines 47-57).

Furthermore, Rusu discloses a cell in queue memory consists of two parts, data and a control header. The control header is established by the queue controller and contains the link list pointers (or address) of the data, as illustrated in Figure 7. As shown in Figure 7, a detailing control memory (145), the header portion (Q1, F1), may be stored in one queue memory location and the tail portion may be stored in another queue memory. The link list is used to enable this operation and keeps track of the mapping relationships (Rusu, col. 4, lines 58-67).

The Final Office Action alleged that Rusu discloses "a packet transmit engine...capable of receiving an address in a queue; reading packet data from an entry from a first data structure with the same address as the received address...reading a next address in the queue from the received address in the queue; and using the next address to repeat the reading packet data and the transmitting," citing Figure 2, box 202 output arbiter, column 4, lines 62-67, Figure 6, box 460, and column 5, lines 31-34 (See Final Office Action at pages 11-12).

However, a review of these passages and figures demonstrates that Rusu fails to disclose or suggest the aforementioned features recited in claim 17. For example, Rusu, at column 4, lines 62-67, fails to disclose or suggest, "a packet transmit engine...capable of receiving an address in a queue" (emphasis added). Rather, Rusu, as previously discussed above, merely discloses a cell in queue memory consisting of two parts, data and a control header. The control header is established by the queue controller and contains the link list pointers (or address) of the data, as illustrated in Figure 7. The header portion of a data cell may be stored in one queue memory location and the tail portion may be stored in another queue memory location. *The link list, or address, of the data enables this operation.* Hence, no *address* is received in the queue; rather, the link list, or the address, is used to keep track of and map relationships. Therefore, the queue controller's output arbitration subsystem (202) is not capable of receiving and address, e.g. the subsystem (202) does not receive the link lists. Thus, Rusu fails to disclose or suggest, at least, "a packet transmit engine...capable of receiving an address in a queue," as recited in claim 17 (emphasis added).

Accordingly, Rusu also fails to disclose or suggest, at least, "a packet transmit engine...capable of reading packet data from an entry from a first data structure with the same address as the received address...reading a next address in the queue from the received address in the queue; and using the next address to repeat the reading packet data and the transmitting," as recited in claim 17.

Therefore, for at least the reasons discussed above, Appellant respectfully submits that Rusu fails to disclose or suggest every feature recited in claim 17.

Accordingly, Appellant respectfully asserts that the Final Office Action failed to establish a *prima facie* case of anticipation as Rusu fails to disclose each and every element of claim 17. As such, reconsideration and withdrawal of the rejection of claim 16 is respectfully requested.

For all of the above noted reasons, it is strongly contended that certain clear differences exist between the present invention as claimed in claims 1-17 and the prior art relied upon by the Examiner. It is further contended that these differences are more than sufficient that the present invention would not have been obvious to a person having ordinary skill in the art at the time the invention was made.

This final rejection being in error, therefore, it is respectfully requested that this honorable Board of Patent Appeals and Interferences reverse the Examiner's decision in this case and indicate the allowability of application claims 1-17.

In the event that this paper is not being timely filed, Appellant respectfully petitions for an appropriate extension of time. Any fees for such an extension together with any additional fees which may be due with respect to this paper may be charged to Counsel's Deposit Account 50-2222.

Respectfully submitted,

SQUIRE, SANDERS & DEMPSEY LLP

Brad Y. Chin
Attorney for Appellant
Registration No. 52,738

Atty. Docket No.: 058268.00325

8000 Towers Crescent Drive, 14th Floor
Tysons Corner, VA 22182-6212
Tel: (703) 720-7800
Fax (703) 720-7802

BYC:dlh

Encls: Appendix 1 - Claims on Appeal
Appendix 2 - Evidence
Appendix 3 - Related Proceedings

APPENDIX 1

CLAIMS ON APPEAL

1. (Original) A method, comprising:

receiving a packet;

determining an address of a free entry in a queue;

placing the determined address in an entry of a prior-determined address in the queue to form a linking list; and

placing packet data of the packet in a free entry of a first data structure,

wherein there is one-to-one mapping between the queue and the first data structure.

2. (Original) The method of claim 1, wherein the packet is unicast.

3. (Original) The method of claim 1, wherein the packet is multicast or broadcast, and further comprising:

determining an address of a free entry in each queue associated with a destination in the packet; and

for each queue associated with a destination in the packet, placing the respective determined address in a respective entry of a prior-determined address in each respective queue.

4. (Original) The method of claim 1, further comprising:

determining a priority level for the received packet; and wherein the placing the determined address places the determined address in an entry of a prior-determined address in the queue having the same priority.

5. (Original) The method of claim 4, wherein the determining a priority level includes examining a quality of service field within the received packet.

6. (Original) The method of claim 1, further comprising updating free entry data to indicate that the determined address is in use.

7. (Original) The method of claim 1, further comprising placing a packet length of the packet in a free entry of a second data structure; and wherein there is one-to-one mapping between the first data structure and the second data structure.

8. (Original) A transmit queue system, comprising:

means for receiving a packet;

means for determining an address of a free entry in a queue;

means for placing the determined address in an entry of a prior-determined address in the queue to form a linking list; and

means for placing packet data of the packet in a free entry of a first data structure,

wherein there is one-to-one mapping between the queue and the first data structure.

9. (Original) A transmit queue system, comprising:

a first data structure capable of holding a plurality of packet data;

a queue capable of holding a linking list of addresses, the addresses having a one-to-one mapping with addresses in the first data structure;

a packet receiving engine capable of receiving a packet;

a free entry engine coupled to the packet receiving engine and capable of determining an address of a free entry in the queue;

a transmit queue engine, coupled to the queue, the packet receiving engine and the free entry engine, capable of placing the determined address in an entry of a prior-determined address in the queue to form a linking list; and

a packet buffer engine, coupled to the first data structure, the packet receiving engine and the free entry engine, capable of placing packet data of the packet in a free entry of the first data structure.

10. (Original) A method, comprising:

receiving an address in a queue;

reading packet data from an entry from a first data structure with the same address as the received address, the queue and the first data structure having one-to-one mapping;

transmitting the packet data to a network node associated with the queue;

reading a next address in the queue from the received address in the queue; and

using the next address to repeat the reading packet data and the transmitting.

11. (Original) The method of claim 10, further comprising reading packet

length from a second data structure, the second data structure having one-to-one mapping with the first data structure.

12. (Original) The method of claim 10, wherein the receiving receives an address for higher priority packet data before receiving an address for lower priority packet data.

13. (Original) The method of claim 10, wherein the packet data is multicast or broadcast and the transmitting transmits the packet data to a plurality of network nodes.

14. (Original) The method of claim 10, wherein the packet data is unicast.

15. (Original) The method of claim 10, further comprising updating free entry data to indicate an address is free after the transmitting.

16. (Original) A transmit queue system, comprising:

means for receiving an address in a queue;

means for reading packet data from an entry from a first data structure with the same address as the received address, the queue and the first data structure having one-to-one mapping;

means for transmitting the packet data to a network node associated with the queue;

means for reading a next address in the queue from the received address in the queue; and

means for using the next address to rerun the means for reading packet data and the means for transmitting.

17. (Original) A transmit queue system, comprising:

a first data structure holding a plurality of packet data;

a queue holding a linking list of addresses, the addresses having a one-to-one mapping with addresses in the first data structure; and

a packet transmit engine, coupled to the first data structure and the queue, capable of

receiving an address in the queue,

reading packet data from an entry from the first data structure with the same address as the received address,

transmitting the packet data to a network node associated with the queue,

reading a next address in the queue from the received address in the queue, and

using the next address to repeat the reading packet data and the transmitting.

APPENDIX 2

EVIDENCE APPENDIX

No evidence under section 37 C.F.R. 1.130, 1.131, or 1.132 has been entered or will be relied upon by Appellant in this appeal.

APPENDIX 3

RELATED PROCEEDINGS APPENDIX

No decisions of the Board or of any court have been identified under 37 C.F.R. §41.37(c)(1)(ii)